



Team Controller  
Northern Arizona University  
Flagstaff, Arizona  
October 20th, 2023

Zachary Parham (Team Lead): [zjp29@nau.edu](mailto:zjp29@nau.edu)  
Italo Santos (Mentor): [ids37@nau.edu](mailto:ids37@nau.edu)  
Bradley Essegian: [bbe24@nau.edu](mailto:bbe24@nau.edu)  
Brandon Udall: [bcu8@nau.edu](mailto:bcu8@nau.edu)  
Dylan Motz: [djm658@nau.edu](mailto:djm658@nau.edu)

Requirements Specification  
Northrop Grumman  
Weapon System Support Software  
Harlan Mitchell  
Laurel Enstrom

Accepted as baseline requirements for this project by:

Client : \_\_\_\_\_

Team Lead : \_\_\_\_\_

Signature

Date

## **Terminology**

**System** - All files developed for the purposes of satisfying the client level requirements.

- **Data display module** - displays status updates and events to the user
- **Controller simulator** - Generates status updates and events then sends via RS422 serial protocol to the data display module.
- **Event Log file** - Will be generated by the data display module to contain 1 or more event strings encountered during a session.
- **Installer** - will deploy the project in the customers system and perform environment setup and initialization. Will record initial user preferences and take them into account during installation. I.e. “Do you want a shortcut on your desktop?”

**Environment** - The directory our system will be placed in and all of its contents.

**Serialized data** - A string of 1s and 0s which can be translated to traditional data such as strings and integers. Only serialized data can be sent through serial ports

**Serial Port** - The physical hardware port which can send and receive serialized data.

**Status data** - (See class diagram for specifics) General data pertaining to the weapon which can be measured at any point in time during session.

**Event String** - A string of the format “<time> <event message> <param 1> <param 2> <param 3>” generated by the controller simulator to simulate the occurrence of a weapon related event. The parameters can be NULL, but the event message must be specific text outlining what the event is. Ex. “[00:12:41] Weapon overheat 237 200” where 237 represents measured barrel temp and 200 represents max recommended barrel temp in degrees celsius.

**Session** - The time measured from the moment the controller sim is connected to the data display module to the moment the controller sim is disconnected from the data display module.

**Handshake Protocols** - A necessary set of agreements between two devices before they perform serial communication.

**Electrical data** - A data structure containing float values for current and voltage and a name for the component.

**Command Line Interface** - method of interacting with programs on a computer by inputting text commands /data/ prompts.

**Automatic Log File Generation** - The ability for the system to automatically generate an event log file after a session has ended.

### **Traceability Matrix**

The purpose of this table is to define which system requirements satisfy each client requirement and which software requirements/functions will be used by each system requirement.

<b>Client requirement</b>	<b>System requirements</b>	<b>Software requirements</b>	<b>Test Case</b>
<b>CR01</b> The data display module shall be a desktop application.	<b>R01</b> The data display module shall be an .exe file.	NA	NA
	<b>R02</b> The data display module shall display a GUI.	SR15 - SR19	NA
<b>CR02</b> The data display module shall read input data via RS422 serial protocol from the controller simulator.	<b>R03</b> The data display module shall be capable of serializing / deserializing messages received via an RS422 serial port.	SR01 - SR05 SR08	<ul style="list-style-type: none"> <li>● test_transmit</li> <li>● test_checkForValidMessage</li> </ul>
	<b>R04</b> The controller simulator shall be	SR01 - SR05 SR08	<ul style="list-style-type: none"> <li>● test_transmit</li> <li>● test_checkForValidMessage</li> </ul>

	capable of serializing / deserializing messages received via an RS422 serial port.		
<b>CR03</b> The data display module shall have the ability to write event data into a log file	<b>R05</b> The data display module shall be capable of generating a log file including all known events when requested by the user	SR11 SR15	<ul style="list-style-type: none"> <li>test_outputToLogFile</li> <li>test_outputToLogFile_badInput_logFileName</li> <li>test_appendToLogFile</li> <li>test_loadDataFromLogFile</li> <li>test_loadDataFromLogFile_badInput_logFileName</li> </ul>
	<b>R06</b> The user shall be able to determine if a log file will be automatically generated after a session.	SR11 SR20	<ul style="list-style-type: none"> <li>test_appendToLogFile</li> </ul>
	<b>R07</b> The user shall be able to determine how many auto saved log files will be kept before overwrites occur on the oldest autosaved file.	SR20	NA
<b>CR04</b> The data display module shall display all weapon status information directly to the application's window for the duration of a session.	<b>R08</b> The controller simulator shall send status updates through the designated serial port every 250 milliseconds (2	SR01 SR05 SR14	<ul style="list-style-type: none"> <li>tst_status::*</li> </ul>

	seconds).		
<b>CR05</b> The controller simulator shall send event updates to the data display module.	<b>R09</b> The controller simulator shall send event updates through the serial port at most 100 milliseconds after they are generated. second after they are generated.	SR01 SR05 SR12 SR13	<ul style="list-style-type: none"> <li>tst_events::*</li> </ul>
<b>CR06</b> The data display module shall not require admin rights to install, set up, or use.	<b>R10</b> The data display module shall not require admin rights to install setup or use	NA	NA
<b>CR07</b> The data display module shall include filtering options to filter events and errors	<b>R11</b> The data display module shall have the capability to display <b>only errors</b> to the Events tab of the GUI	SR13	NA
	<b>R12</b> The data display module shall have the capability to display <b>only cleared errors</b> to the events tab of the GUI.	SR21	NA
	<b>R13</b> The data display module shall have the capability to display <b>only active errors</b> to the events tab of the GUI.	SR21	Na
	<b>R14</b>	SR21	NA

	The data display module shall have the capability to display only <b>non-error events</b> to the events tab of the GUI.		
<b>CR08</b> The system and its environment shall be installed via an installer file.	<b>R15</b> The system and its environment shall be installed via an installer file	NA	NA
<b>CR09</b> The system shall be portable on Windows 10 or 11	<b>R16</b> The system shall be portable on Windows 10 or 11	NA	NA
<b>CR10G</b> The system should be portable on Debian linux distributions	<b>R17G</b> The system should be portable on Debian linux distributions	NA	NA

### **Client level requirements**

- **CR01** - The data display module shall be a desktop application.
- **CR02** - The data display module shall read input data via RS422 serial protocol from the controller simulator through a user specified port.
- **CR03** - The data display module shall have the ability to write event data into a log file
- **CR04** - The data display module shall display all status information directly to the application's window for the duration of a session.
- **CR05** - The controller simulator shall send event updates to the data display module.
- **CR06** - The data display module shall not require admin rights to install, set up, or use.
- **CR07** - The data display module shall include filtering options to filter events and errors
- **CR08** - The system and its environment shall be installed via an installer file.

- **CR09** - The system shall be portable on Windows 10 or 11
- **CR10G** - The system should be portable on Debian linux distributions

## **System Level Requirements**

- **R01** - The data display module shall be an .exe file.
- **R02** - The data display module shall display a GUI.
- **R03** - The data display module shall be capable of serializing / deserializing messages received via an RS422 serial port.
- **R04** - The controller simulator shall be capable of serializing / deserializing messages received via an RS422 serial port.
- **R05** - The data display module shall be capable of generating a log file including all known events when requested by the user
- **R06** - The user shall be able to determine if a log file will be automatically generated after a session.
- **R07** - The user shall be able to determine how many auto saved log files will be kept before overwrites occur on the oldest autosaved file.
- **R08** - The controller simulator shall send status updates through the designated serial port every 250 milliseconds.
- **R09** - The controller simulator shall send event updates through the serial port at most 100 milliseconds after they are generated.
- **R10** - The data display module shall not require admin rights to install setup or use
- **R11** - The data display module shall have the capability to display **only errors** to the Events tab of the GUI
- **R12** - The data display module shall have the capability to display **only cleared errors** to the events tab of the GUI.
- **R13** - The data display module shall have the capability to display **only active errors** to the events tab of the GUI.
- **R14** - The data display module shall have the capability to display only **non-error events** to the events tab of the GUI.
- **R15** - The system and its environment shall be installed via an installer file
- **R16** - The system shall be portable on Windows 10 or 11

- **R17G** - The system should be portable on Debian linux distributions

## **Software Level Requirements**

### **Serial Communication:**

- **SR01** - The software shall be capable of generating *serialized* versions of given *status data* and *event data*.
- **SR02** - The software shall be capable of generating *status data* given *serialized* status data.
- **SR03** - The software shall be capable of generating an *event string* given a *serialized event string*.
- **SR04** - The software shall be capable of generating *electrical data* given *serialized electrical data*.
- **SR05** - The software shall be capable of sending *serialized data* through a *serial port*.
- **SR06** - The software shall be configurable to fill one of the the following roles during *handshake protocols*
  - a. send the first contact message every 5 seconds until a response is received
  - b. listen for the first contact message, then respond.
- **SR07** - *Handshake protocols* shall be implemented using the Boost.Asio serial library
- **SR08** - The software shall be able to listen for and record serialized bit strings from a given *serial port*.
- **SR09** - The software shall be able to pause serial communication
- **SR10** - The software shall be able to resume serial communication



- **SR11** - The software shall be capable of storing all event strings received via serial communication until a new session is started or the program ends.

#### **Controller Simulator:**

- **SR12** - The software shall be capable of generating *event strings* with random *parameters* given a collection of *event messages*.
  - **Note:** See terminology section for definitions of event string, event message and parameters
- **SR13** - The software shall be capable of reading *event strings* from the *command line interface*.
- **SR14** - The software shall be capable of generating randomized *status data*.

#### **Data Display Module:**

- **SR15** - The software shall be capable of writing an event log file in csv format, given a collection of events upon user request.
- **SR16** - The software shall be capable of opening the events page when the events button is pushed.
- **SR17** - The software shall be capable of opening the status page when the status button is pushed.
- **SR18** - The software shall be capable of opening the electrical page when the electrical button is pushed.
- **SR19** - The software shall be capable of opening the connection settings page when the connection settings button is pushed.
- **SR20** - The software shall allow the user to input how many auto saved log files they want to be kept before overwrites occur on the oldest autosaved file.
- **SR21** - The software shall allow the user to input what filter they want on the event page out of the following options.
  - a. Only errors
  - b. Only cleared errors
  - c. Only active errors
  - d. Non error events

